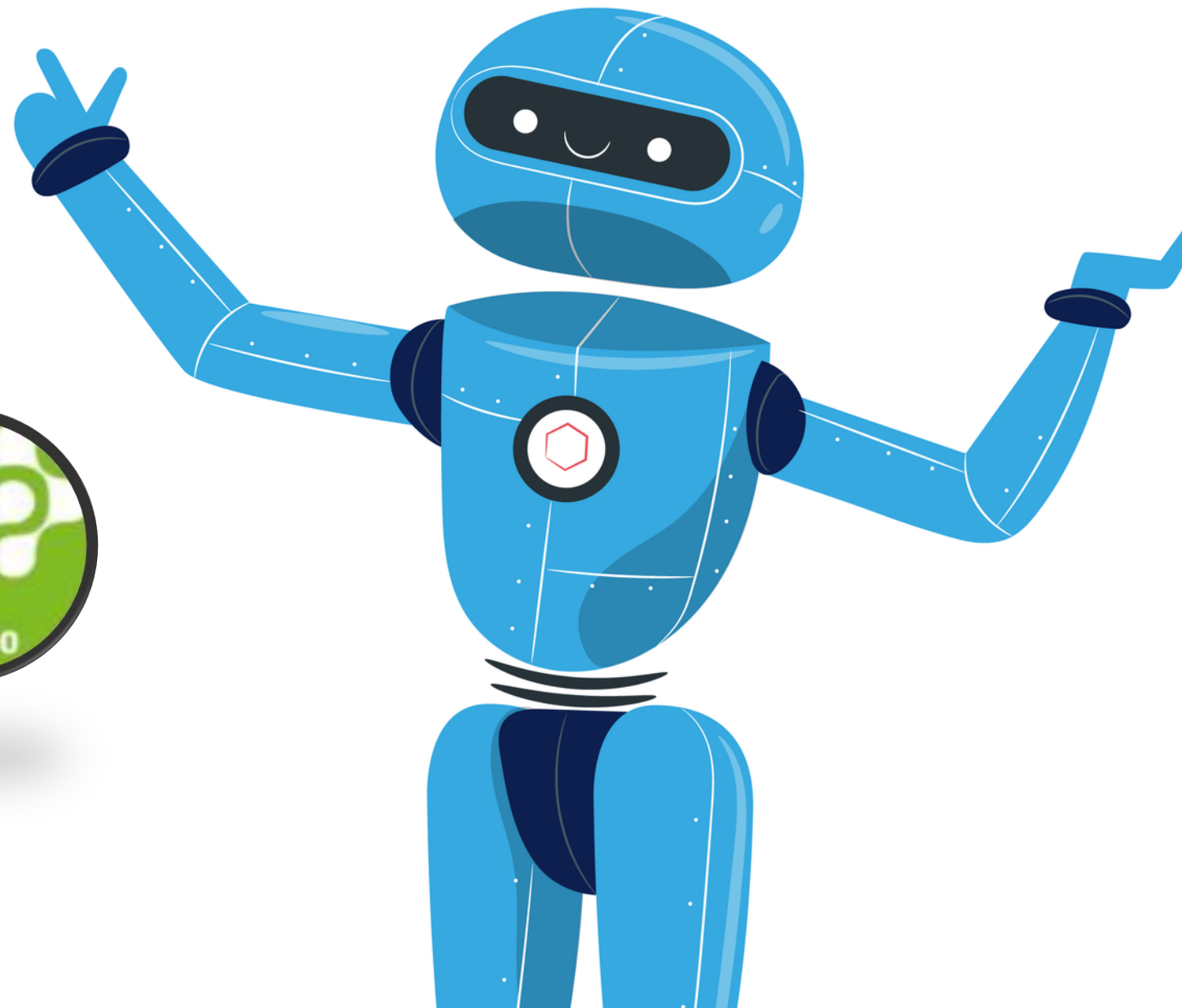


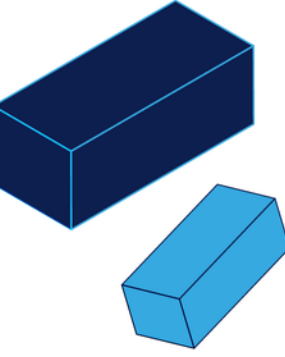
CLASE 0

INTRODUCCION A

MICROCONTROLADORES

ARM





INTRODUCCION A LA ARQUITECTURA ARM

arm

MICRO-
CONTRO-
LADORES
ARM

- **AMR Holding**
 - (Advanced RISC Machine)

- **Desarrolla la arquitectura y da licencia** a otras empresas, para que diseñen sus propios productos que implementan una de esas arquitecturas (**SoC**) y (**SoM**).
- **Diseña núcleos** que implementan conjunto de instrucciones y **otorga licencias** de estos diseños a varias empresas. Que incorporan esos diseños centrales en sus propios productos.

The ARM logo consists of the lowercase letters "arm" in white, centered within a solid blue square.

arm

The Texas Instruments logo features a white outline of the state of Texas on the left, followed by the words "TEXAS INSTRUMENTS" in white, uppercase letters on a red rectangular background.

TEXAS INSTRUMENTS

The ARM Microcontroller logo features the lowercase letters "arm" in blue, centered within a white rounded rectangle that has a blue border.

arm

MICRO-
CONTRO-
LADORES
ARM

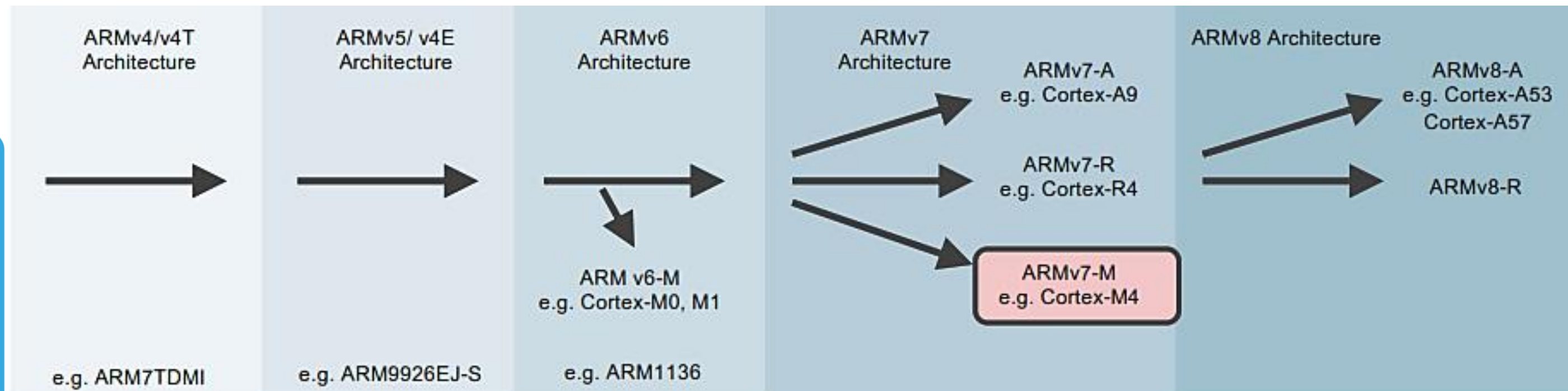
DIFERENCIA ENTRE PROCESADOR ARM Y ARQUITECTURA ARM

- **ARQUITECTURA ARM**

- ✓ Describe los detalles de instrucciones, modelo de programación, interrupciones, mapa de memoria.
- ✓ Esta documentada en el manual de referencia de la arquitectura.

- **PROCESADOR ARM**

- ✓ Desarrollado en base a una de las arquitecturas ARM.
- ✓ Contiene detalles de implementación real, como diagramas de tiempos, mapa concreto de memoria. Su documentación principal es la hoja de características (processor's Technical Reference Manual).



arm

MICRO-
CONTRO-
LADORES
ARM

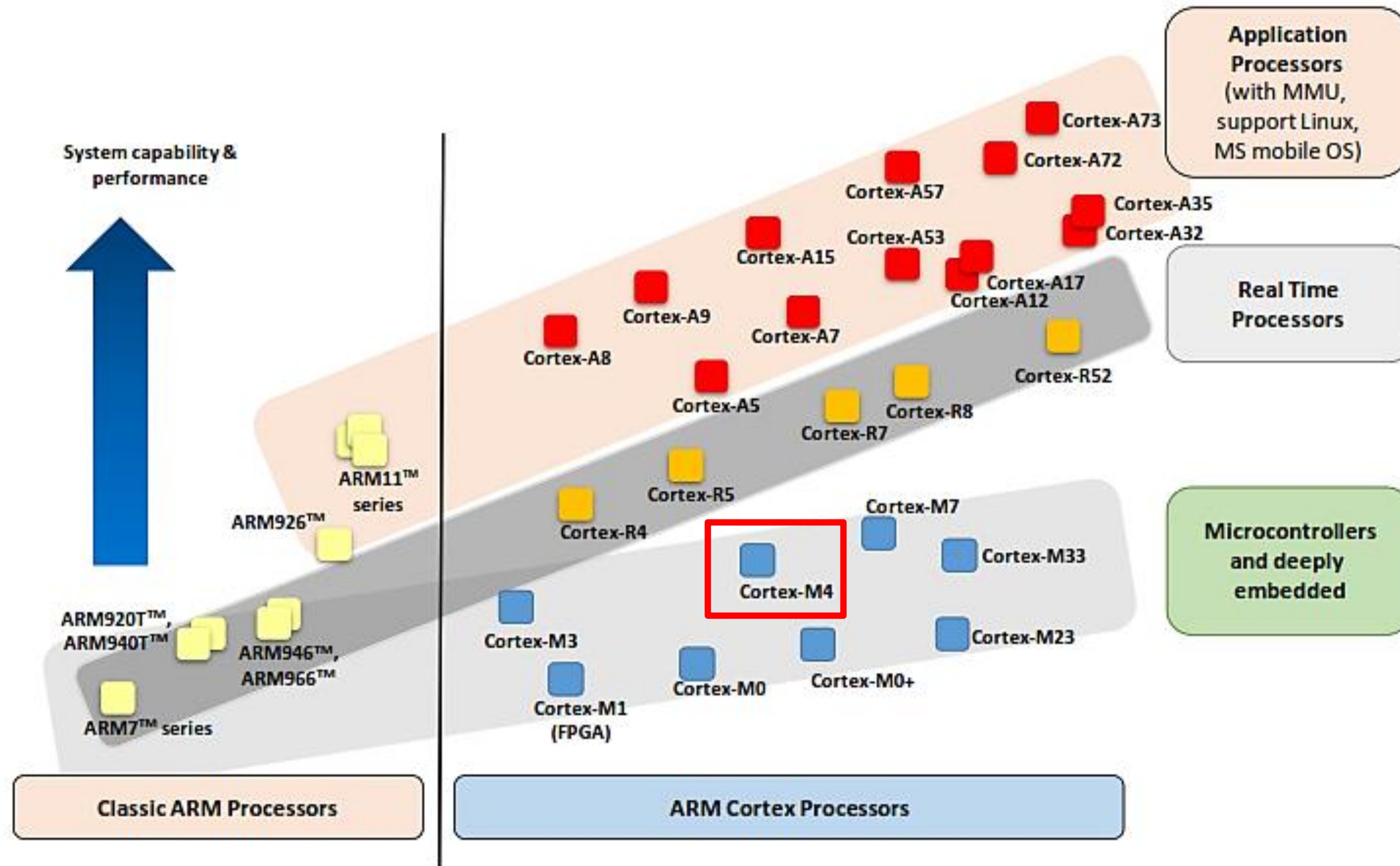
ESPECIFICACIONES DE LA ARQUITECTURA

Instruction set	<ul style="list-style-type: none">• La función de cada instrucción.• Cómo se representa esa instrucción en la memoria (su codificación).
Register set	<ul style="list-style-type: none">• Cuántos registros hay.• El tamaño de los registros.• La función de los registros.• Su estado inicial.
Exception model	<ul style="list-style-type: none">• Los diferentes niveles de privilegio.• Los tipos de excepciones.• Qué sucede al aceptar o regresar de una excepción.
Memory model	<ul style="list-style-type: none">• Cómo se ordenan los accesos a la memoria.• Cómo se comportan las cachés, cuándo y cómo el software debe realizar un mantenimiento explícito.
Debug, trace, and profiling	<ul style="list-style-type: none">• Cómo se establecen y activan los puntos de interrupción.• Qué información se puede capturar con las herramientas de rastreo y en qué formato.

arm

MICRO-
CONTRO-
LADORES
ARM

LA FAMILIA DE PROCESADORES ARM



arm

MICRO-
CONTRO-
LADORES
ARM

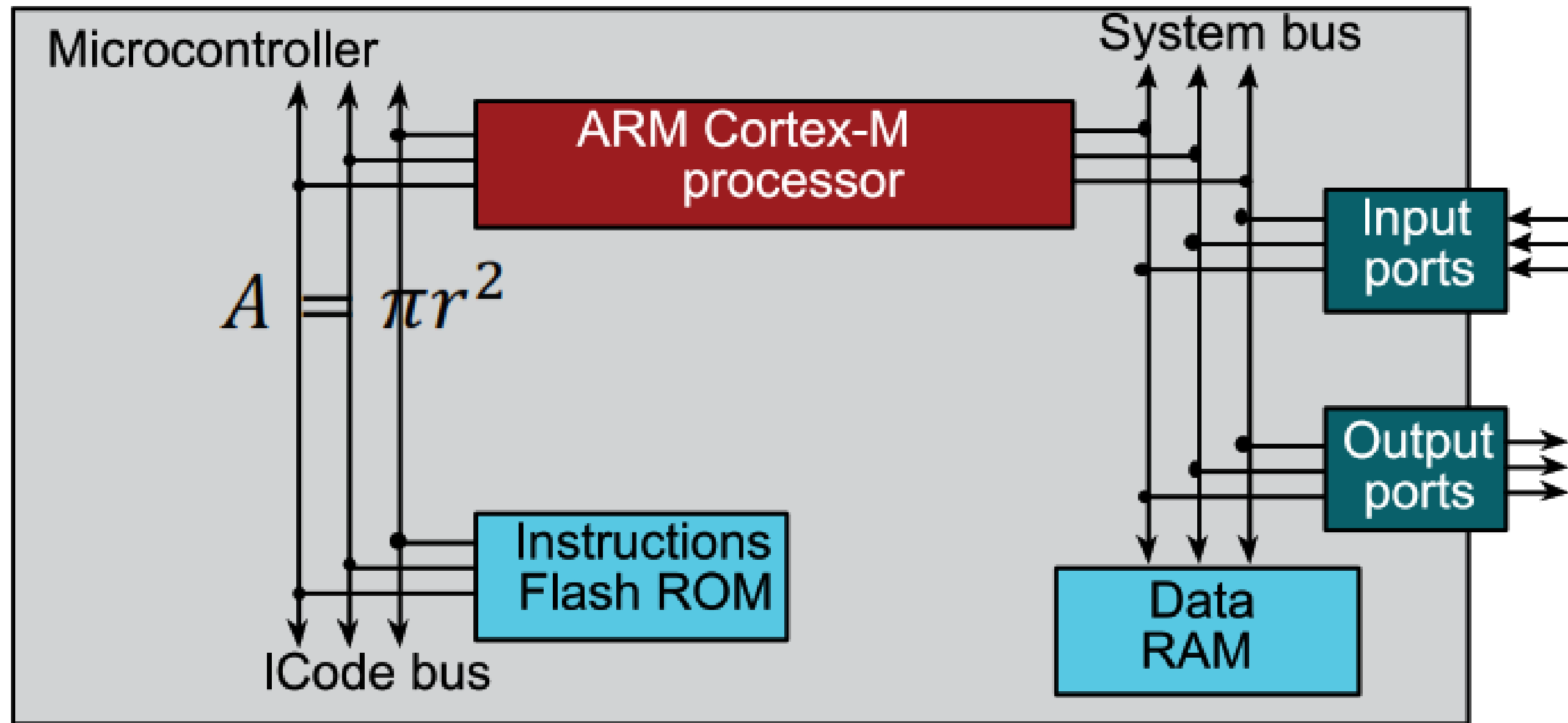
CARACTERISTICAS DE LA ARQUITECTURA ARM

arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

- Arquitectura Harvard vs Von Neuman.
- Diferentes buses para instrucciones y datos.
 - ICode bus → Fetch op codes from ROM
 - System bus → Data from RAM and I/O
 - Dcode bus → Debugging
 - PPB bus → Private peripherals



arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX – M

Reduced Instruction Set Computer (RISC)

CISC	RISC
Many instructions	Few instructions
Instructions have varying lengths	Instructions have fixed lengths
Instructions execute in varying times	Instructions execute in 1 or 2 bus cycles
Many instructions can access memory	Few instructions can access memory <ul style="list-style-type: none">• Load from memory to a register• Store from register to memory
In one instruction, the processor can both <ul style="list-style-type: none">• Read memory and• Write memory	<ul style="list-style-type: none">• No one instruction can both read and write memory in the same instruction
Fewer and more specialized registers <ul style="list-style-type: none">• Some registers contain data• Others contain addresses	<ul style="list-style-type: none">• Many identical general purpose registers
Many different types of addressing modes	Limited number of addressing modes <ul style="list-style-type: none">• Register, PC - relative• Immediate• Indexed

arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

BUS AMBA (Advanced Microcontroller Bus Architecture)

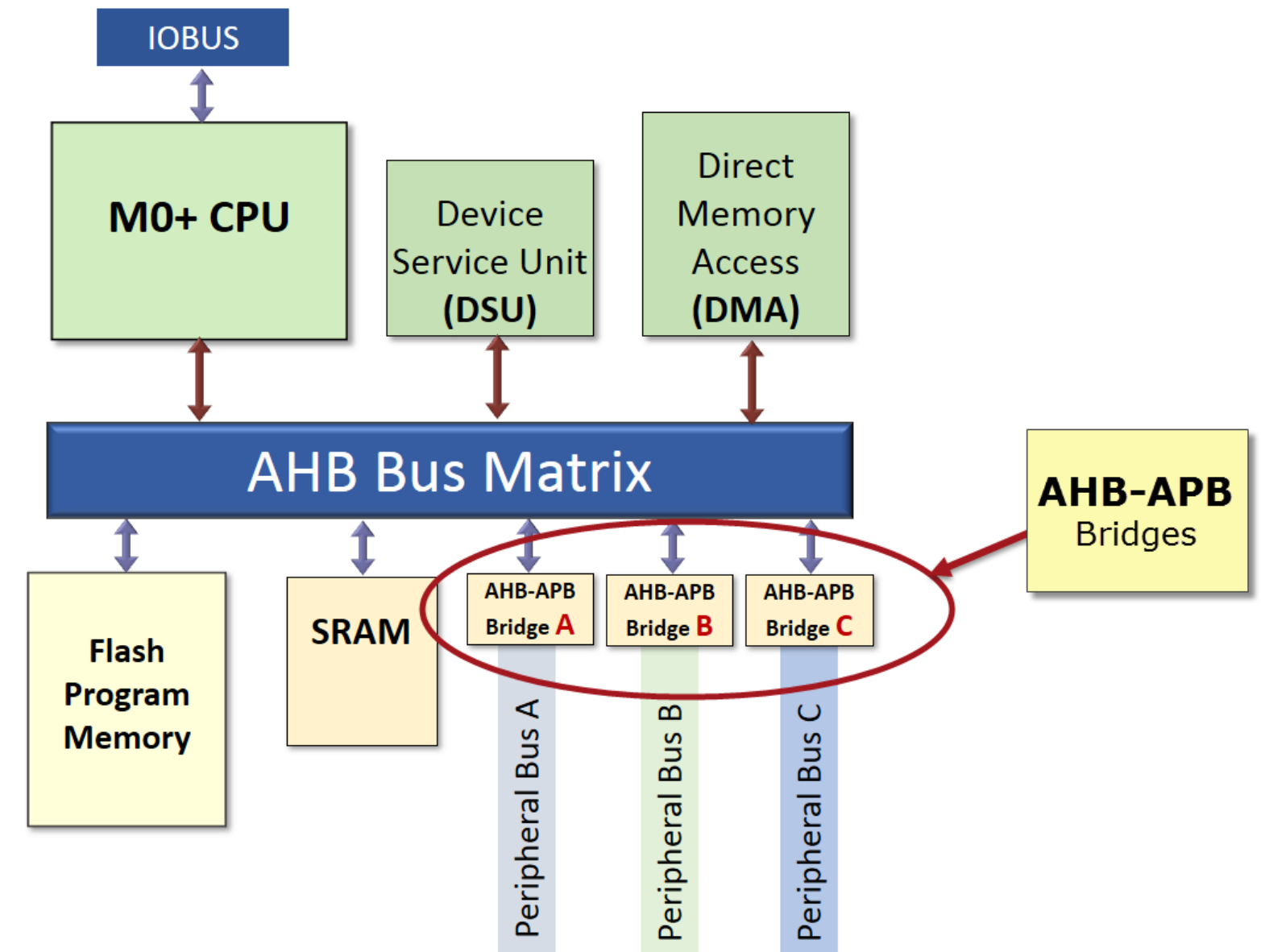
Advanced High-performance Bus (AHB)

- conectar componentes que necesitan mayor ancho de banda en un bus compartido.
- alto rendimiento.
- operación canalizada.
- Transferencia en ráfaga.
- Comunicación maestro esclavo

Advanced Peripheral Bus (APB)

- se utiliza para conectar periféricos de ancho de banda bajo.
- Low power
- Interfaz simple
- Apta para muchos periféricos

Los dos puentes AHB / APB, proporcionan conexiones sincrónicas completas entre el AHB y los dos buses APB, lo que permite una selección flexible de la frecuencia periférica.

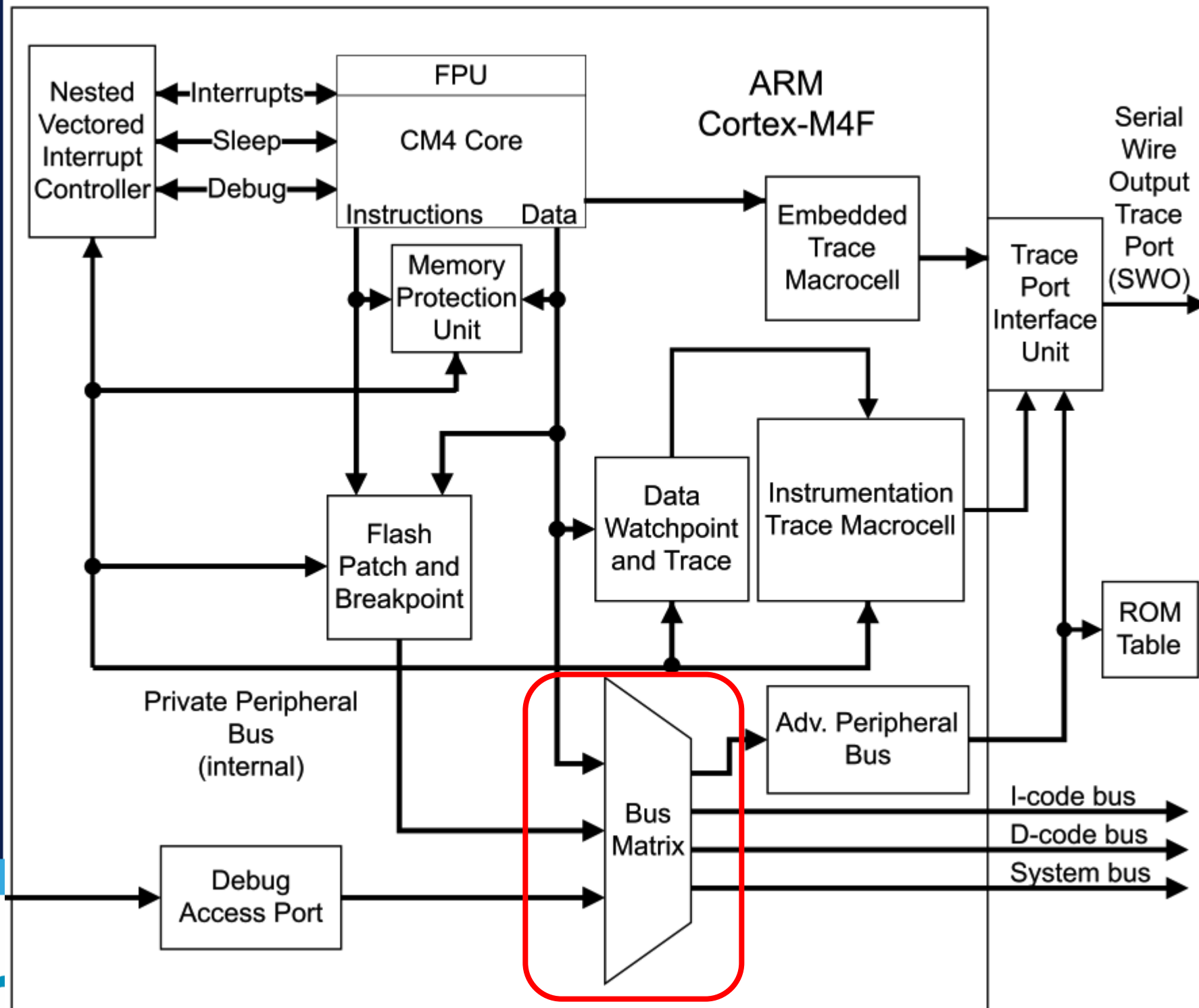


arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

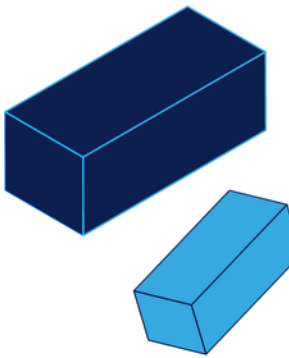
BUS MATRIX



La matriz de bus proporciona acceso de un maestro a un esclavo, lo que permite el acceso concurrente y operación eficiente incluso cuando varios periféricos de alta velocidad funcionan simultáneamente

- **ICode bus** → Fetch op codes from ROM
- **System bus** → Data from RAM and I/O
- **Dcode bus** → Debugging
- **PPB bus** → Private peripherals

PROCESADOR CORTEX - M

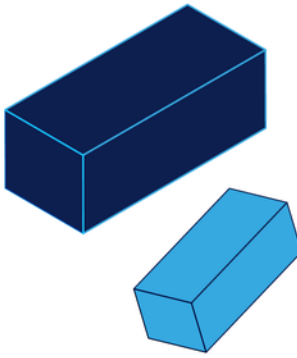


CONJUNTO DE INSTRUCCIONES DE LOS PROCESADORES CORTEX-M

Todos los procesadores Cortex-M admiten un conjunto de instrucciones llamado Thumb. Sin embargo, diferentes procesadores Cortex-M admiten diferentes subconjuntos de las instrucciones disponibles en Thumb ISA

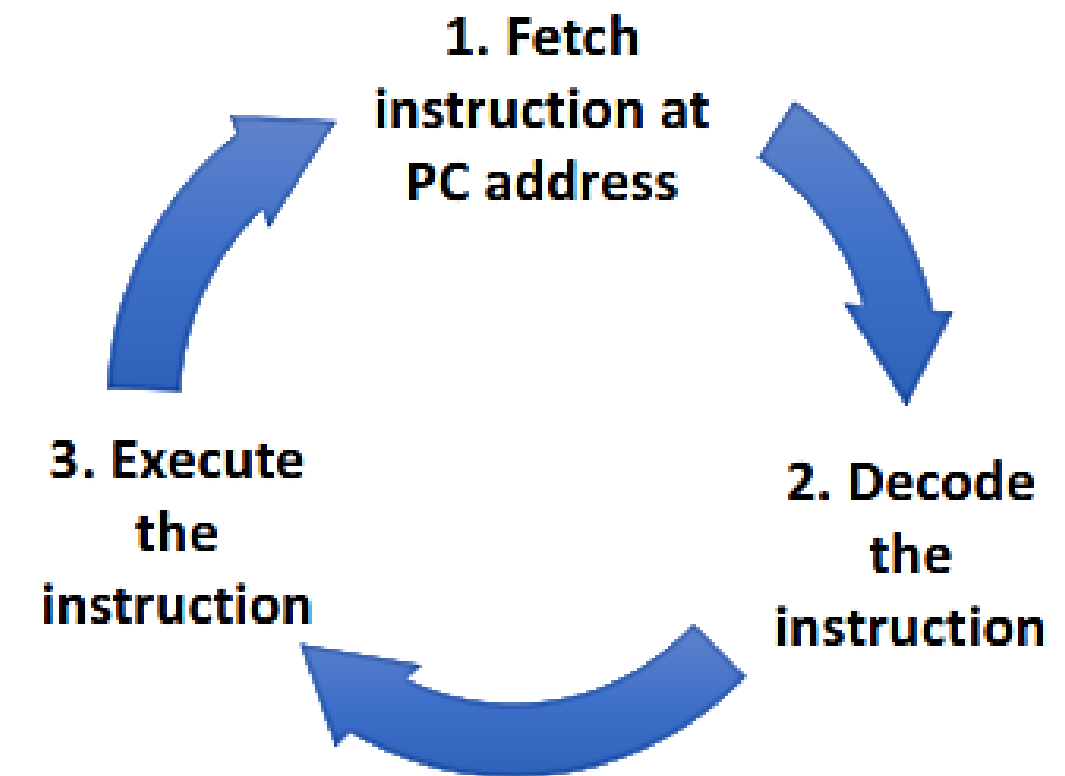
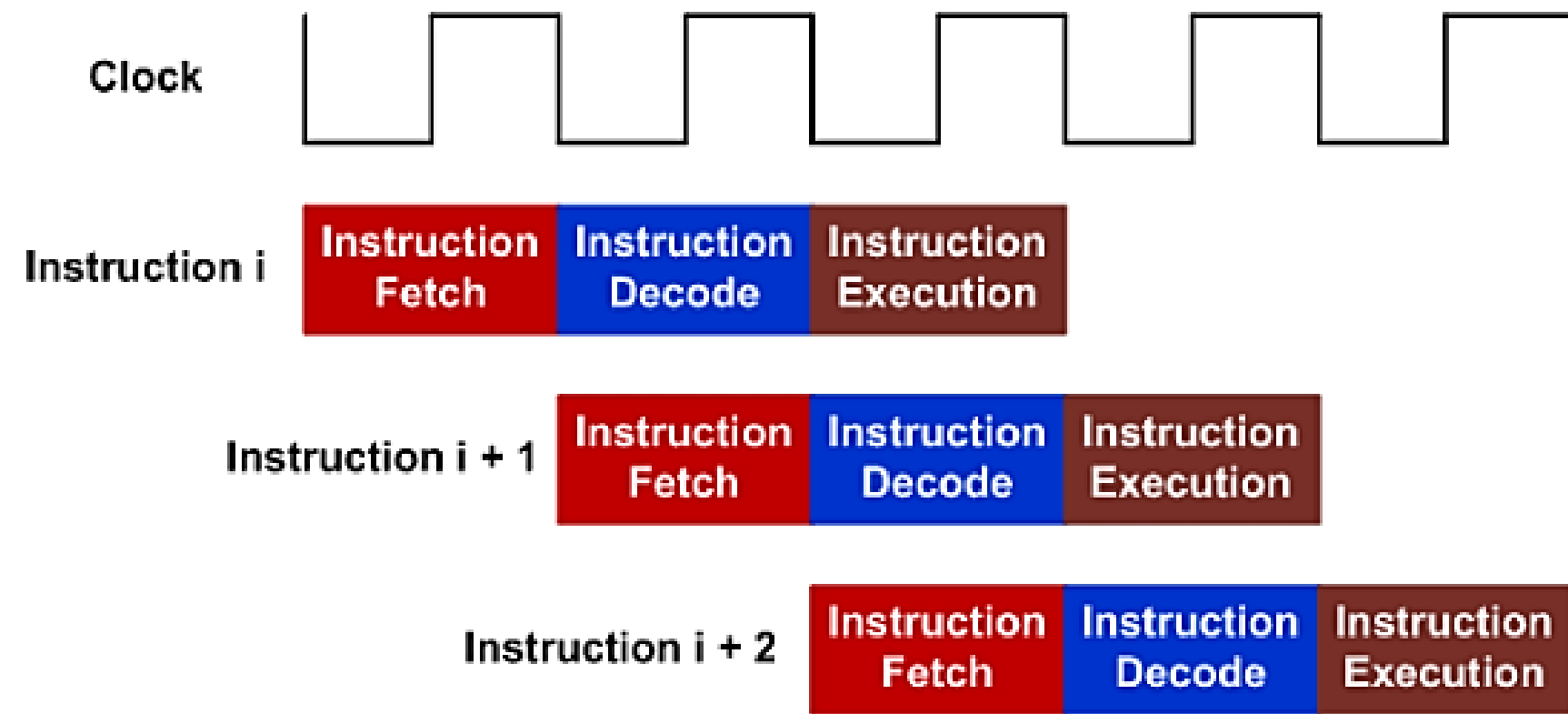


PROCESADOR CORTEX - M



Three-state pipeline: Fetch, Decode, Execution

- La **canalización (Pipelining)** permite que los recursos de hardware se utilicen por completo.
- Se puede obtener una instrucción de 32 bits o dos instrucciones de 16 bits.



arm

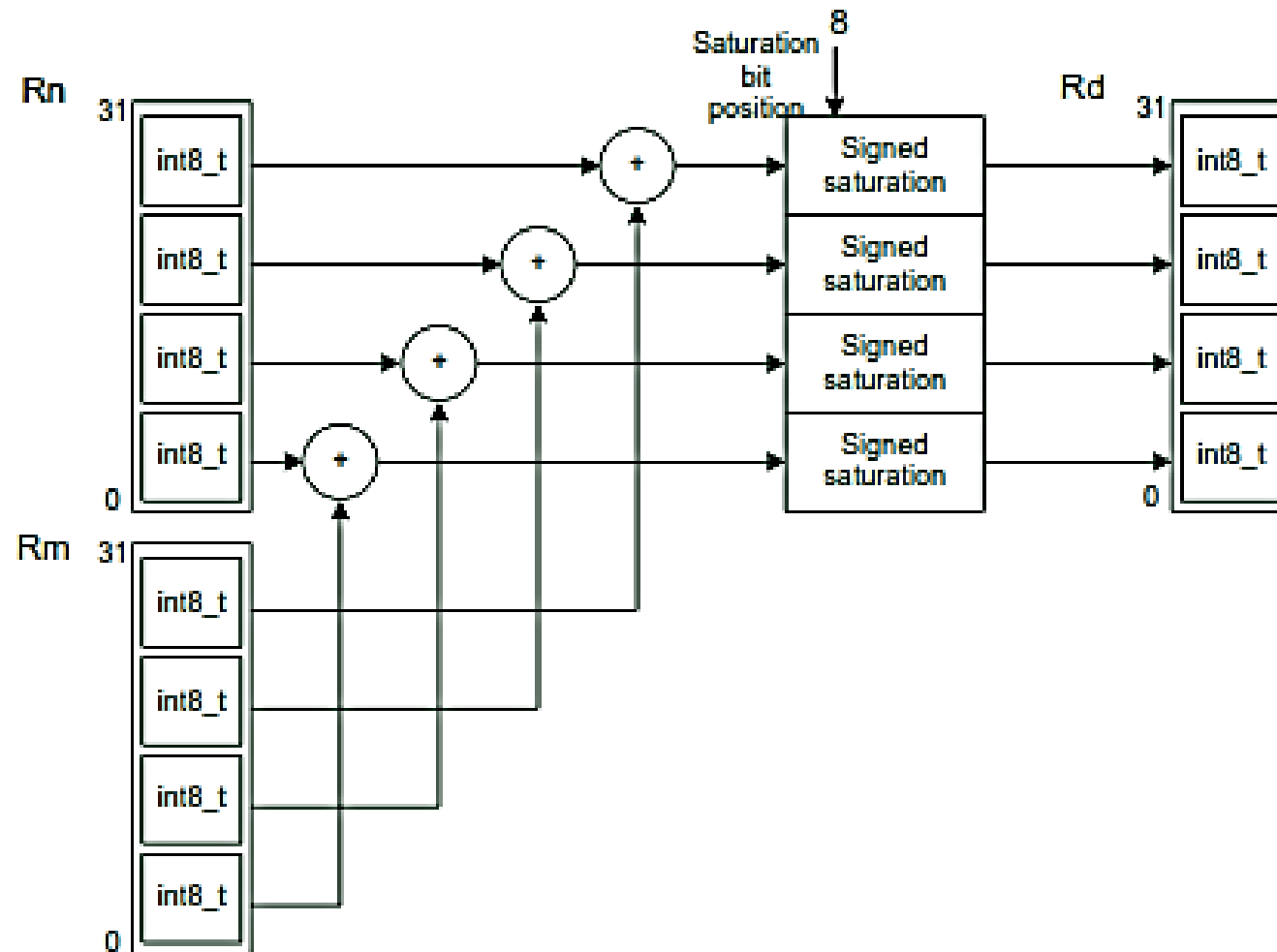
MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

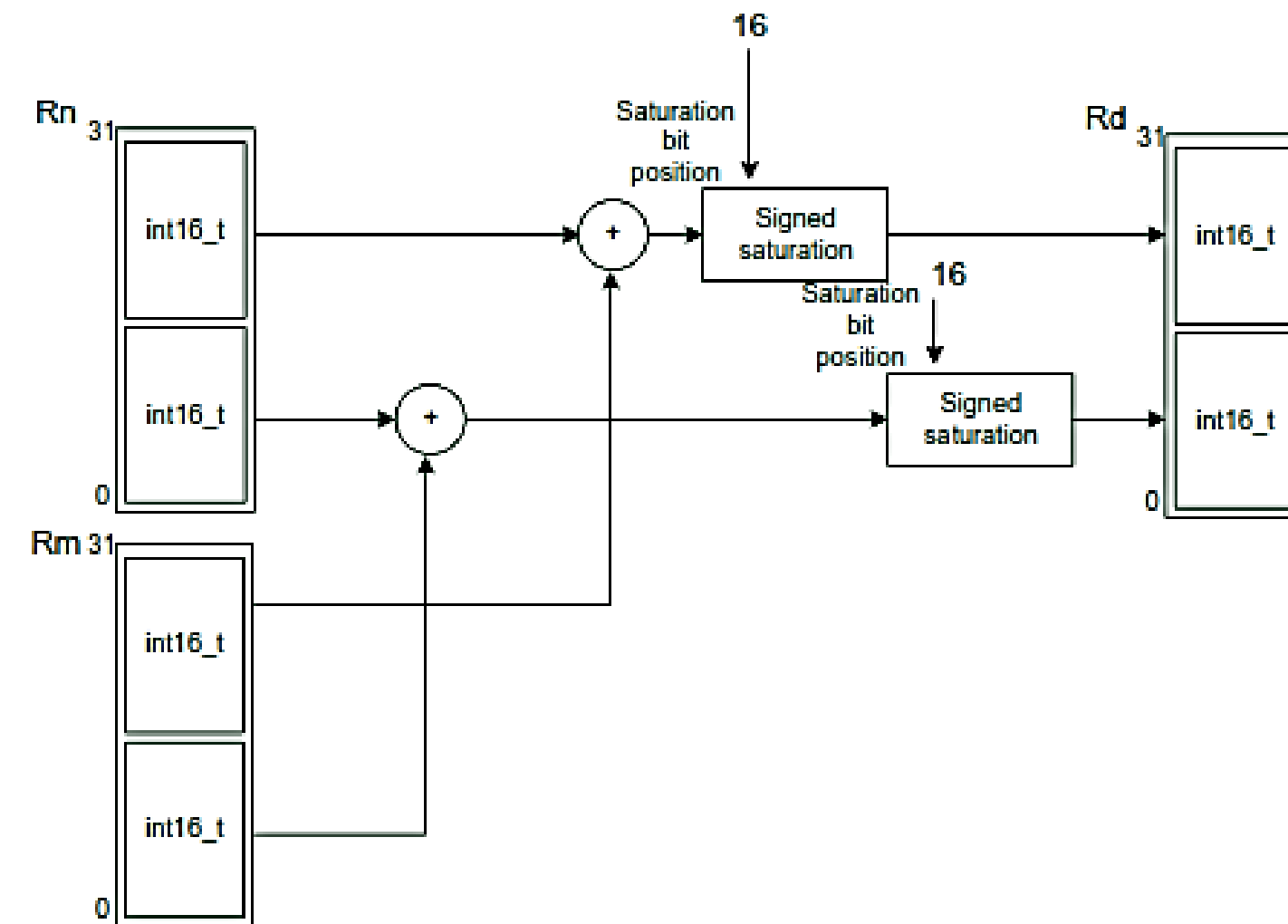
SOPORTE DE INSTRUCCIONES EN CORTEX-M4

El procesador CORTEX-M4 soporta todas las instrucciones CORTEX-M3 y, además admite varias instrucciones destinadas a aplicaciones **DSP** como **SIMD**, instrucciones aritméticas de saturación, una amplia gama de instrucciones **MAC** que se pueden ejecutar en ciclos únicos y una unidad flotante opcional que admite operaciones de punto flotante de precisión simple.

QADD8 {<Rd>} <Rn>, <Rm>



QADD16 {<Rd>} <Rn>, <Rm>



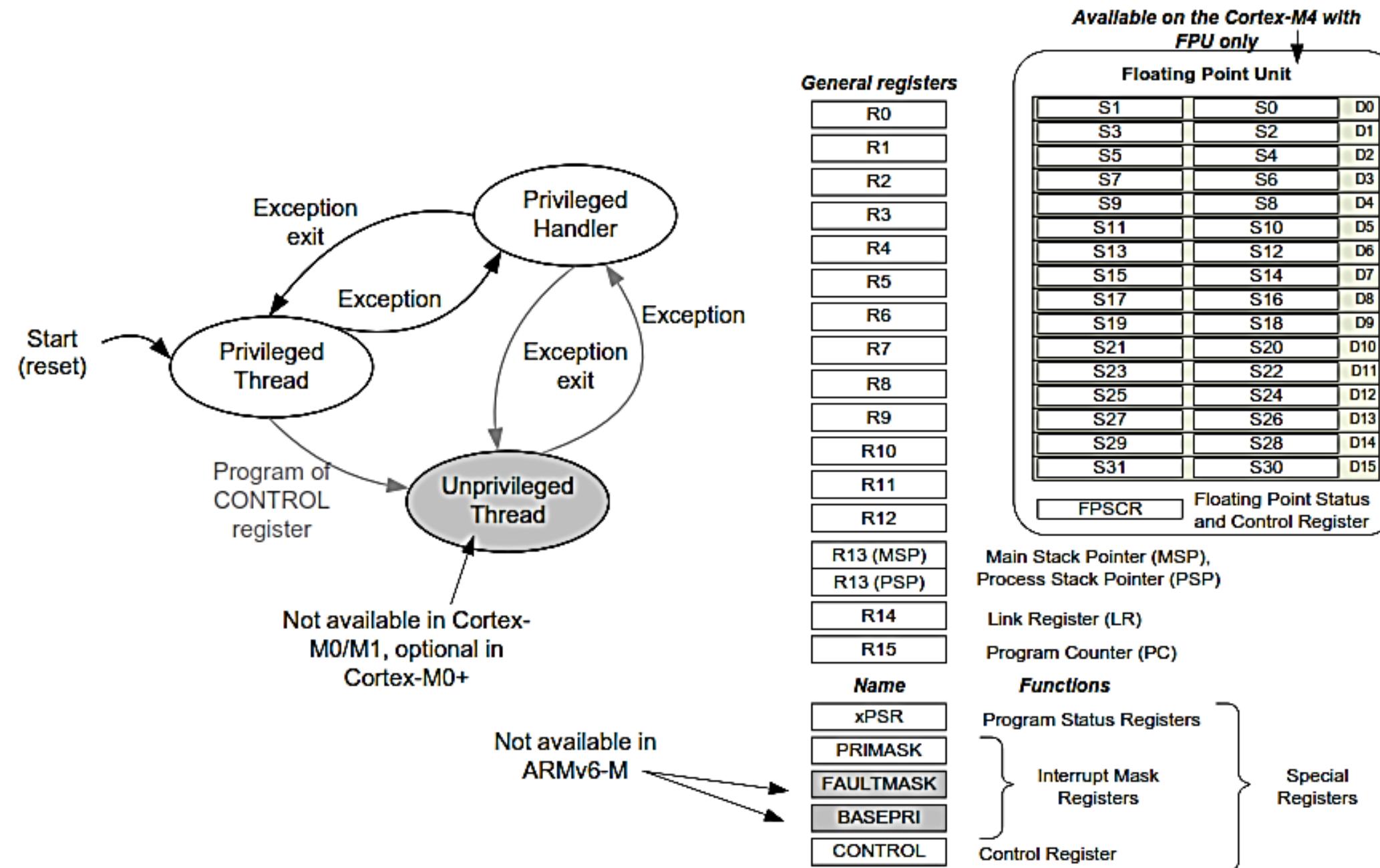
arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

Programmer's model

Los registros R0 a R15, PSR, CONTROL y PRIMASK están disponibles para todos los procesadores Cortex-M. Dos registros especiales **FAULTMASK** y **BASEPRI** están disponibles solo en Cortex-M3, Cortex-M4, Cortex-M7 y Cortex-M33, y el banco de registros de coma flotante y **FPSCR** está disponible en Cortex -M4 / M7 / M33 dentro de la unidad de coma flotante opcional.



arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

CONTROL REGISTER

		31:4	3	2	1	0
ARMv6-M	CONTROL				SPSEL	nPRIV
Cortex-M3	CONTROL				SPSEL	nPRIV
Cortex-M4/M7	CONTROL			FPCA	SPSEL	nPRIV
Cortex-M23	CONTROL				SPSEL	nPRIV
Cortex-M33	CONTROL		SFPA	FPCA	SPSEL	nPRIV

Indicates floating point context is Secure
(for TrustZone security extension)

Indicates floating point context is active

Stack pointer selection for Thread mode

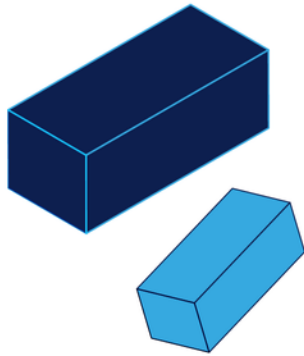
Thread mode unprivileged

arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX - M

PROGRAM STATUS REGISTER



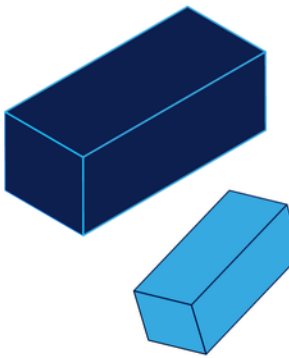
Exception Number

	31	30	29	28	27	26:25	24	23:20	19:16	15:10	9	8	7	6	5	4:0
ARMv6-M (Cortex-M0/M0+)	N	Z	C	V			T									
ARMv7-M (Cortex-M3)	N	Z	C	V	Q	ICI/IT	T			ICI/IT			Exception Number			
ARMv7E-M (Cortex-M4/M7)	N	Z	C	V	Q	ICI/IT	T		GE[3:0]	ICI/IT			Exception Number			
ARMv8-M Baseline (Cortex-M23)	N	Z	C	V			T						Exception Number			
ARMv8-M Mainline (Cortex-M33)	N	Z	C	V	Q	ICI/IT	T		GE[3:0]	ICI/IT			Exception Number			

arm

MICRO-
CONTRO-
LADORES
ARM

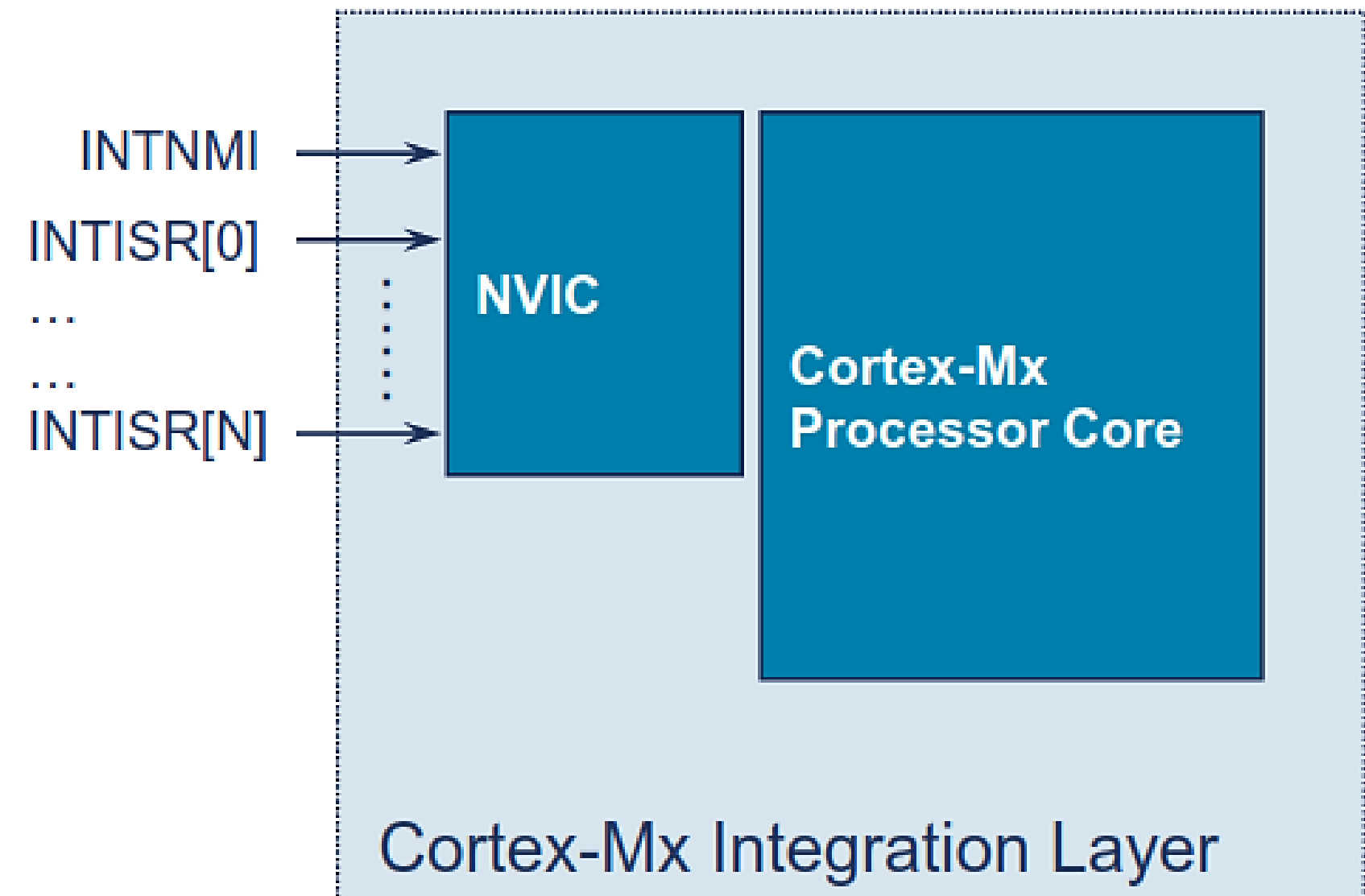
PROCESADOR CORTEX - M



Exception model and NVIC

Todos los procesadores Cortex-M incluyen Controlador de interrupción vectorial anidado (NVIC) y comparten el mismo modelo de excepción.

- Reset
- Non-maskable Interrupts
- Faults
- PendSV
- SVCall
- External Interrupt
- SysTick Interrupt



arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX – M

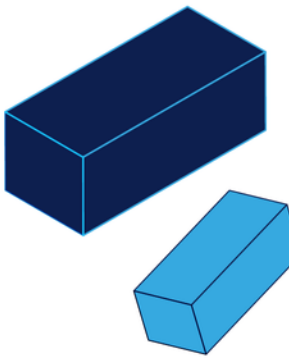
Exception model and NVIC

Exception Type	ARMv6-M (Cortex-M0/M0+/M1)	ARMv7-M (Cortex-M3/M4/M7)	ARMv8-M Baseline (Cortex-M23)	ARMv8-M Mainline (Cortex-M33)	Vector Table	Vector address offset (initial)
495		Not supported in Cortex-M3/M4/M7	Not supported in Cortex-M23		Interrupt#479 vector	10x000007BC
256						
255					Interrupt#239 vector	10x000003FC
31						
17	Device Specific Interrupts	Device Specific Interrupts	Device Specific Interrupts	Device Specific Interrupts	Interrupt#31 vector	10x000000BC
16						
15	SysTick	SysTick	SysTick	SysTick	Interrupt#1 vector	10x00000044
14	PendSV	PendSV	PendSV	PendSV	Interrupt#0 vector	10x00000040
13					SysTick vector	10x0000003C
12	Not used	Not used	Not used	Not used	PendSV vector	10x00000038
11	SVC	SVC	SVC	SVC	Not used	0x00000034
10					Debug Monitor vector	10x00000030
9					SVC vector	10x0000002C
8					Not used	0x00000028
7	Not used	Not used	Not used	Not used	Not used	0x00000024
6		Usage Fault		Usage Fault	Not used	0x00000020
5		Bus Fault		Bus Fault	SecureFault (ARMv8-M Mainline)	10x0000001C
4		MemManage (fault)		MemManage (fault)	Usage Fault vector	10x00000018
3	HardFault	HardFault	HardFault	HardFault	Bus Fault vector	10x00000014
2	NMI	NMI	NMI	NMI	MemManage vector	10x00000010
1					HardFault vector	10x0000000C
0					NMI vector	10x00000008
					Reset vector	10x00000004
					MSP initial value	0x00000000

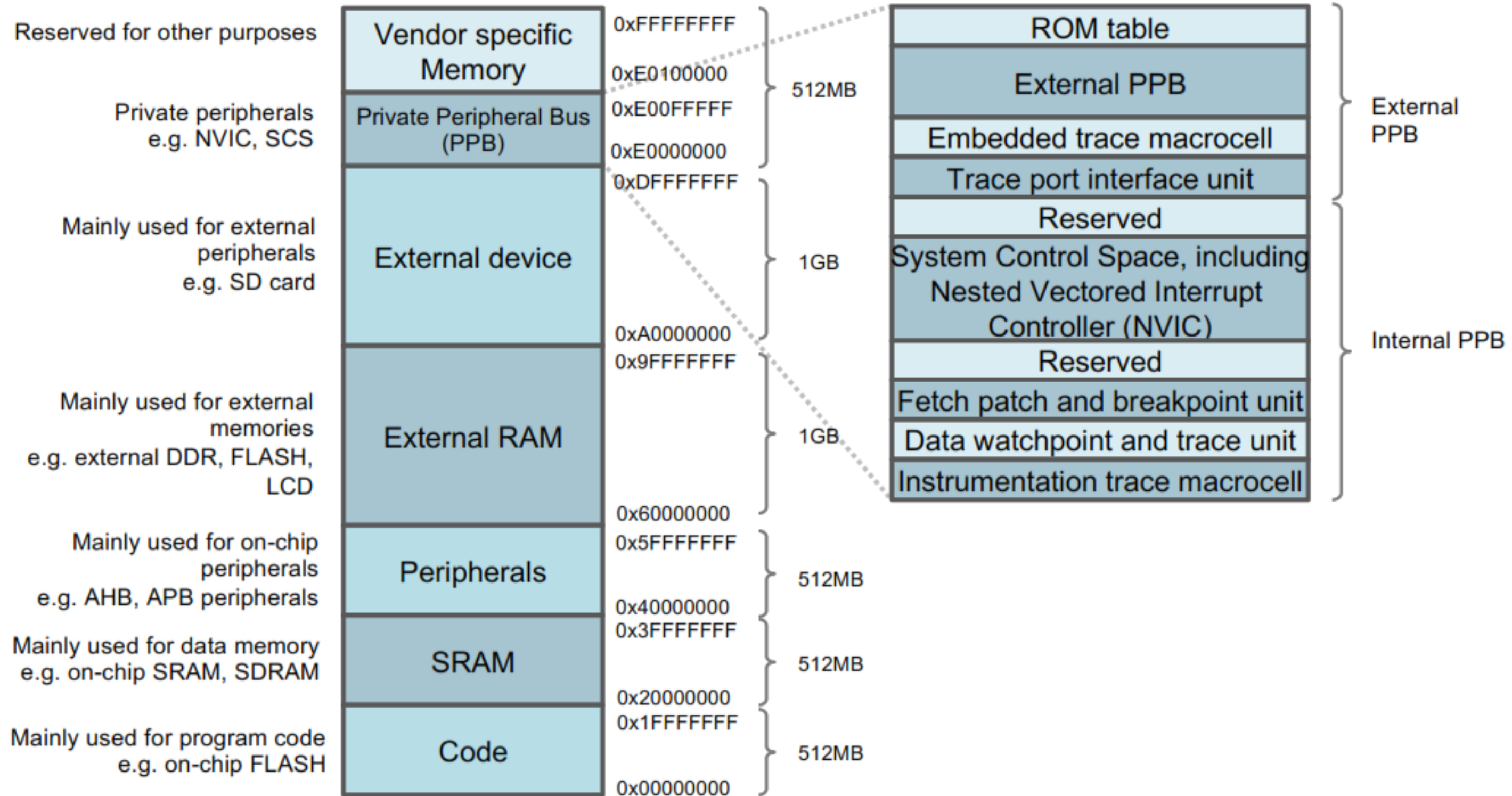
arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX – M



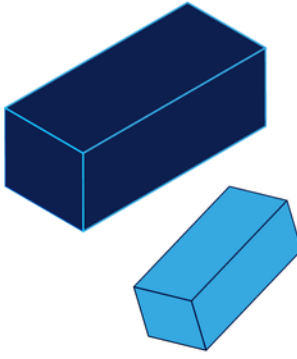
Memory MODEL



arm

MICRO-
CONTRO-
LADORES
ARM

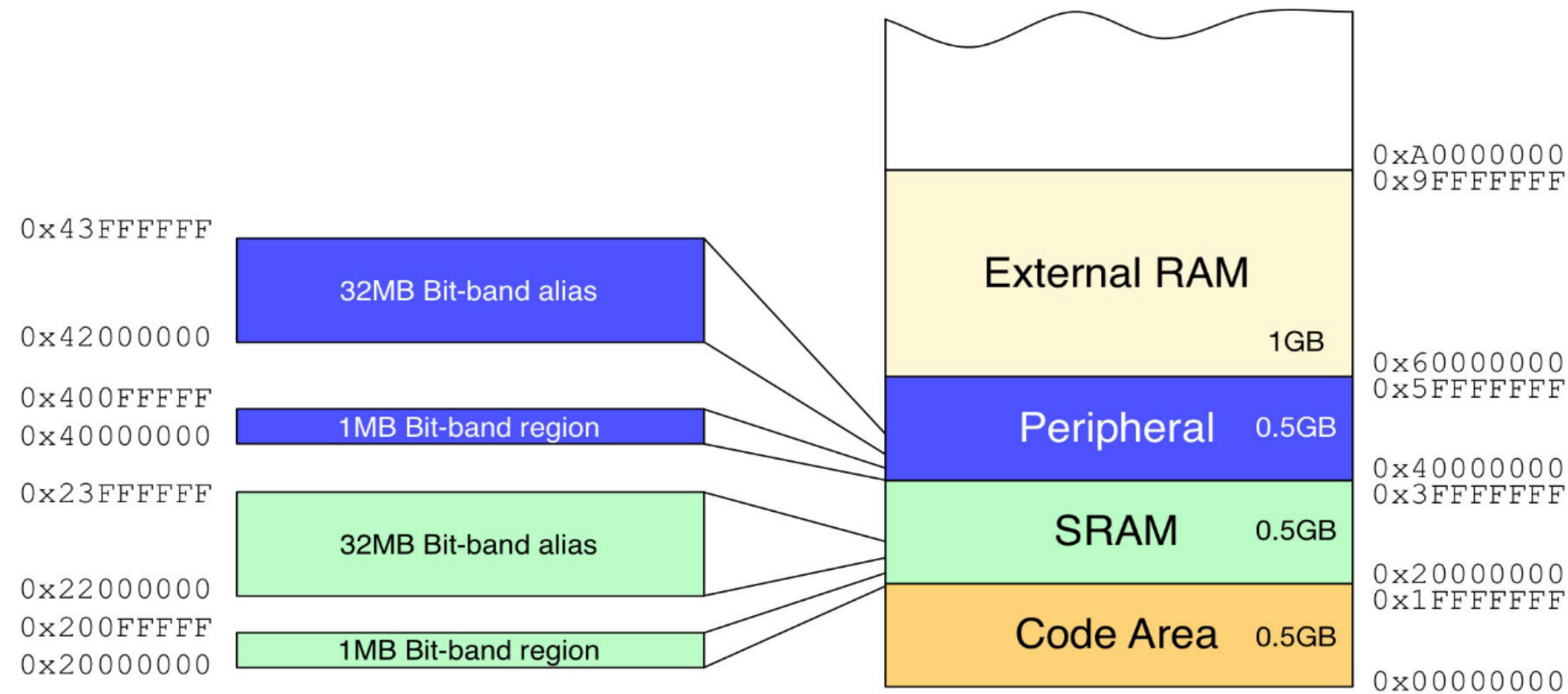
PROCESADOR CORTEX - M



Memory MODEL: Bit Band

Los procesadores Cortex-M3 y Cortex-M4 tienen una característica opcional llamada banda de bits que permite que dos rangos de direcciones de 1 MB (uno en SRAM, desde 0x20000000, el otro en Peripheral, desde 0x40000000) sean direccionables mediante bits a través de direcciones de alias de banda de bits.

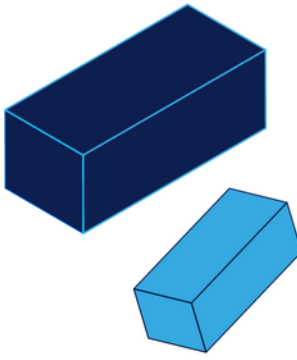
- Bit-Banding es un método para realizar modificaciones atómicas bit a bit en la memoria.



arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX – M



Memory MODEL: Bit Band

- Una fórmula de mapeo muestra cómo hacer referencia a cada palabra en la región alias a un bit correspondiente, o bit de destino, en la región BITBAND. La fórmula de mapeo es:

$$\text{bit_word_offset} = (\text{byte_offset} \times 32) + (\text{bit_number} \times 4)$$

$$\text{bit_word_addr} = \text{bit_band_base} + \text{bit_word_offset}$$

Donde:

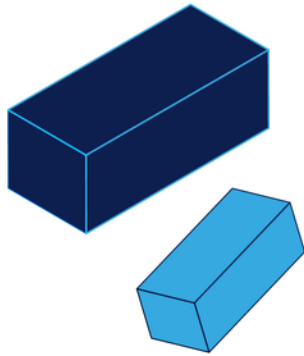
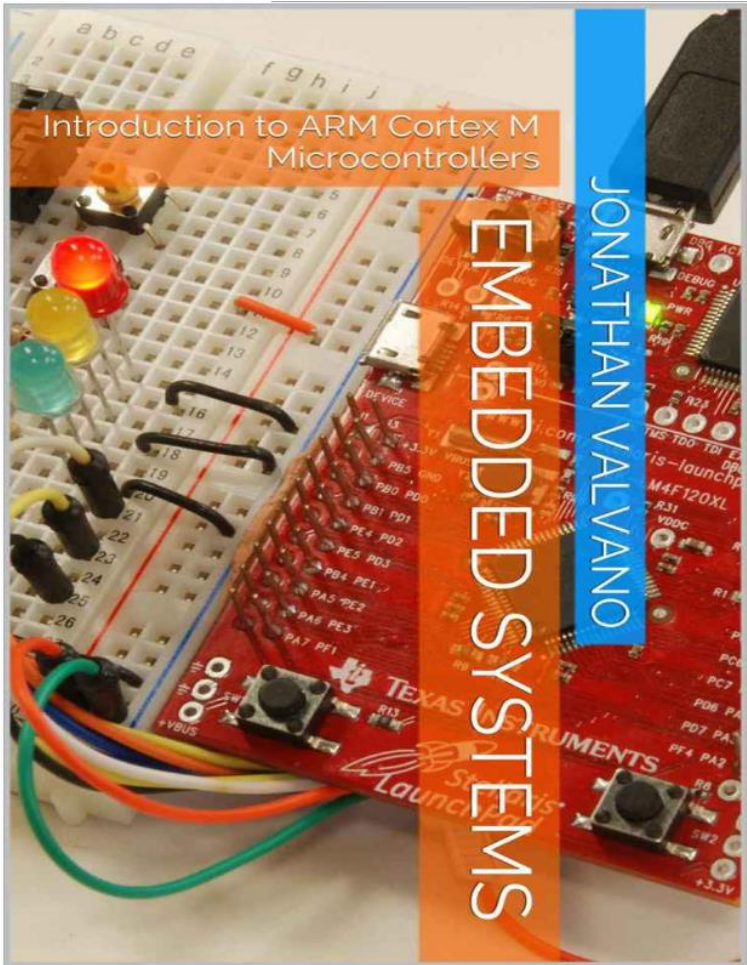
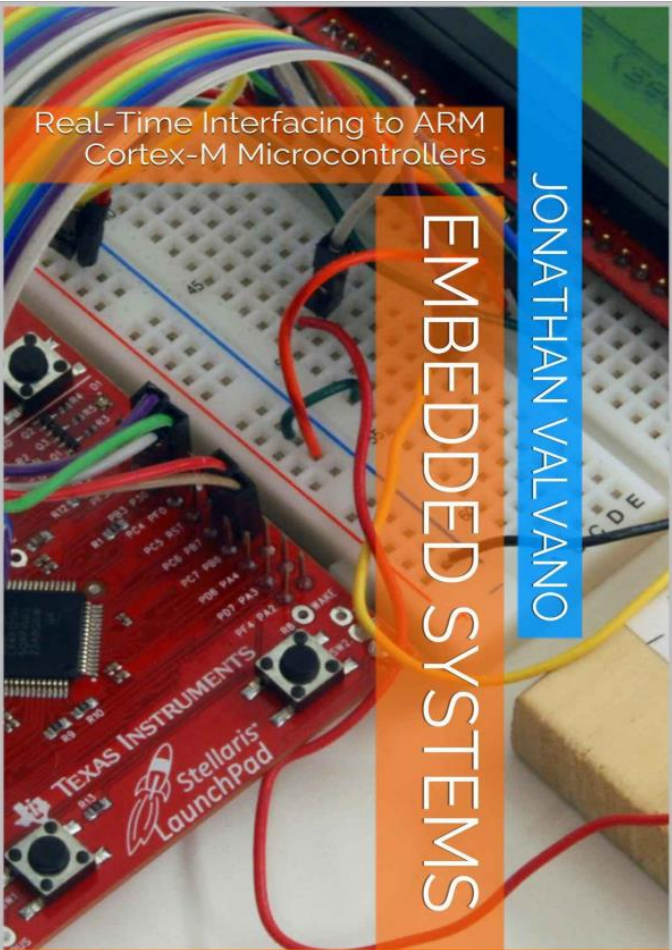
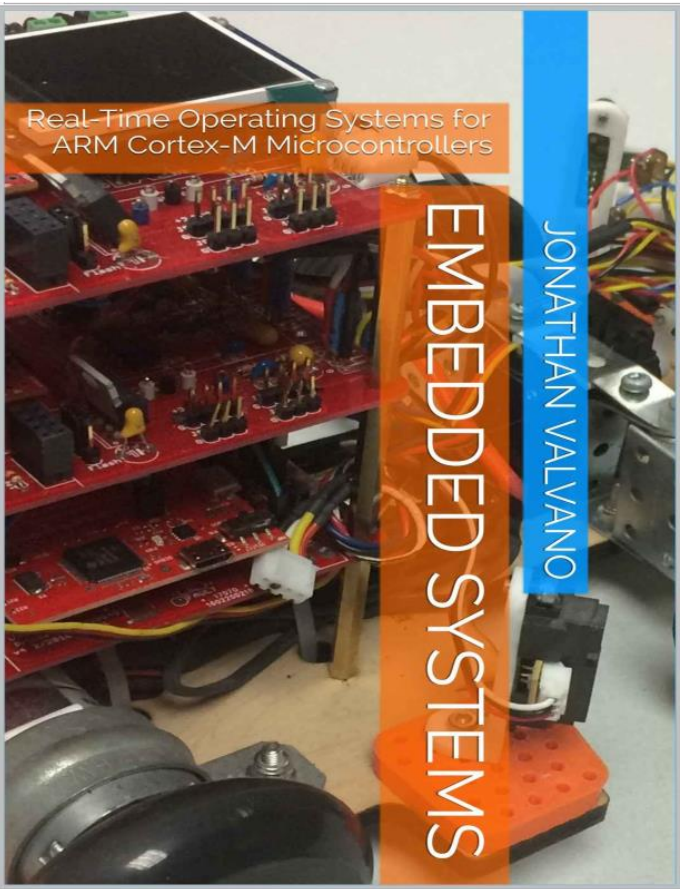
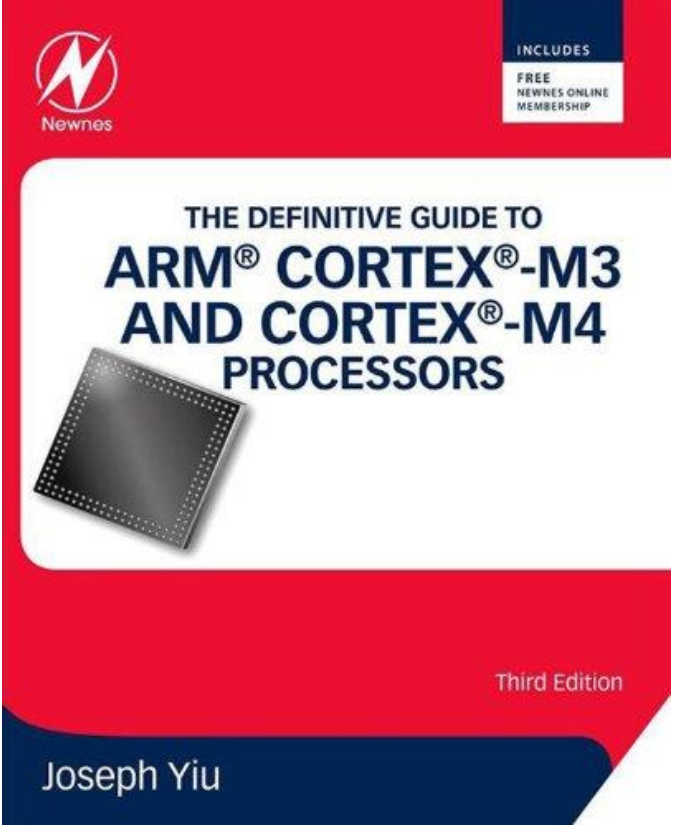
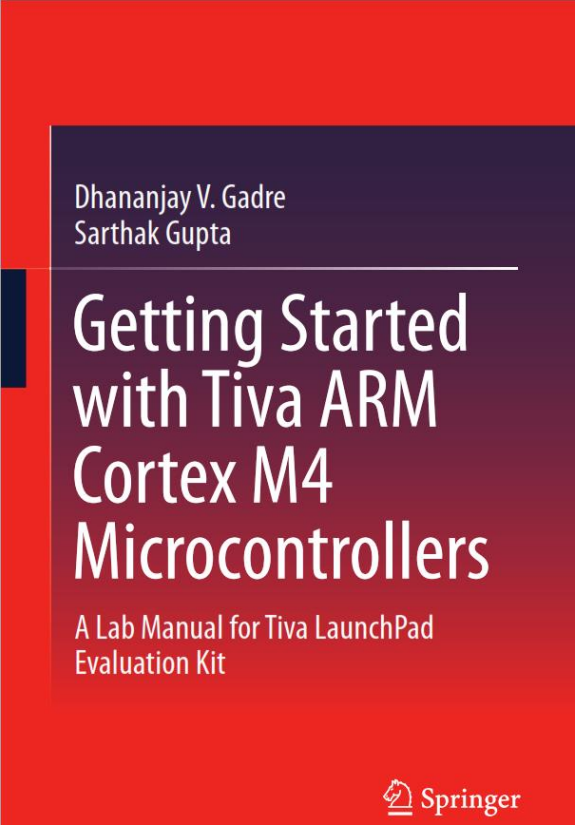
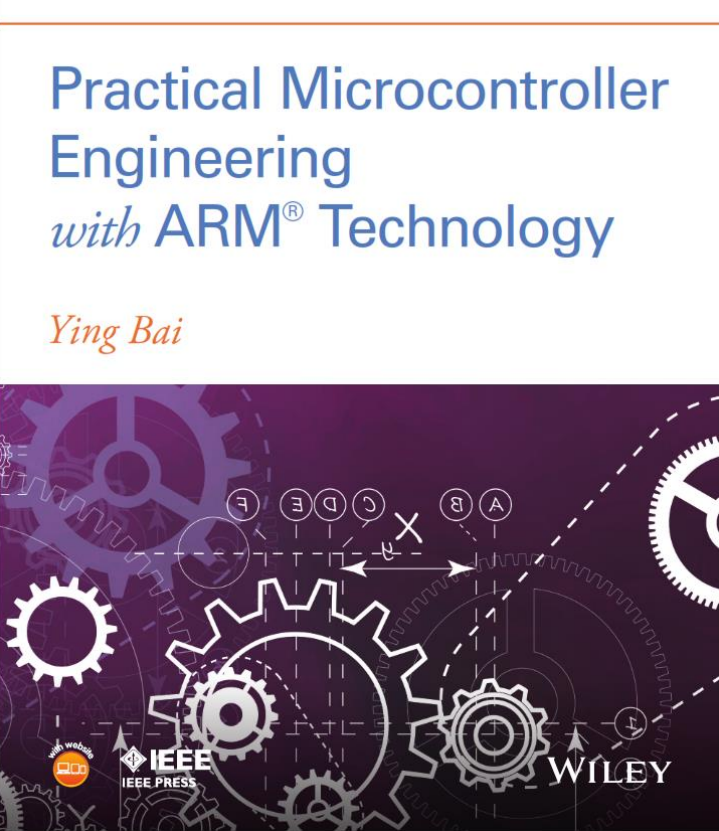
- ✓ **bit_word_offset**, es la posición del bit de destino en la región de memoria bit-band.
- ✓ **bit_word_addr**, es la dirección de la palabra en la región de memoria de alias que se asigna al bit de destino.
- ✓ **bit_band_base**, es la dirección inicial de la región de alias
- ✓ **byte_offset** es el numero del byte en la región bit-band que contiene el bit objetivo.
- ✓ **bit_number** es el bit posición, 0 a 7, de el bit objetivo.

arm

MICRO-
CONTRO-
LADORES
ARM

PROCESADOR CORTEX – M

REFERENCIAS



UMAKER | CENTRO DE CAPACITACIÓN
DE DESARROLLO TECNOLÓGICO